Computers and Mathematics with Applications 60 (2010) 2260–2269

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Optimization of activity coefficient models to describe vapor–liquid equilibrium of (alcohol + water) mixtures using a particle swarm algorithm

Juan A. Lazzús *

Departamento de Física, Universidad de La Serena, Casilla 554, La Serena, Chile

ARTICLE INFO

Article history:
Received 3 February 2010
Received in revised form 5 August 2010
Accepted 6 August 2010

Keywords:
Optimization
Particle swarm optimization
Distillation process
Activity coefficient models
Vapor–liquid equilibrium

ABSTRACT

A method to model the vapor–liquid phase based on a particle swarm algorithm is developed in this study. Two activity coefficient models (UNIQUAC and NRTL) were optimized with particle swarm optimization (PSO), and used to describe the isobaric vapor–liquid equilibrium of fifteen binary mixtures containing alcohol + water. The results were compared with the Levenberg–Marquardt algorithm, and show that the PSO algorithm is a good method to correlate and predict the vapor–liquid equilibrium of this type of system.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

The aim of optimization is to determine the best-suited solution to a problem under a given set of constraints. Mathematically an optimization problem involves a fitness function describing the problem, under a set of constraints representing the solution space for the problem. The optimization problem, nowadays, is represented as an intelligent search problem, where one or more agents are employed to determine the optima on a search landscape, representing the constrained surface for the optimization problem [1].

Because of the difficulties in evaluating the first derivatives, to locate the optima for many rough and discontinuous optimization surfaces, in recent times, several derivative free optimization algorithms have emerged [2]. Particle swarm optimization (PSO) is a relatively recently devised population-based stochastic global optimization algorithm [3]. As described by Eberhart and Kennedy, the PSO algorithm is an adaptive algorithm based on a social-psychological metaphor; a population of individuals (referred to as particles) adapts by returning stochastically toward previously successful regions [4].

Parameter optimization procedures are very important in engineering, industrial, and chemical processes for the development of mathematical models and advanced control of processes [5]. Distillation is a liquid–vapor separation process in which heat is used as the separating agent, process that can be done in a batch or continuous manner. Fig. 1 shows the typical batch distillation processes used with common mixtures of alcohol + water. In this equipment, the mixture to be separated, the must contained in the pot still, is heated to its boiling temperature, at which point the boiling liquid and the vapor produced are at thermodynamic equilibrium. Separation is done in the distillation column in which the vapor going to the top of the column becomes richer in the more volatile components, the heavier components still living in the pot. The lighter components are then converted into liquid in the condenser. Part of this liquid forms the distilled product and part of it is recycled to the column to get into contact with the rising vapor [7]. Therefore, knowledge of vapor–liquid equilibrium data (VLE) is necessary to design and optimize distillation processes.

* Tel.: +56 51 204128; fax: +56 51 206658.
E-mail address: jlazzus@dfuls.cl.

0898-1221/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2010.08.016

Notation

Symbols

B_{ij}	NRTL parameter
c_1	Cognitive acceleration constant
c_2	Social acceleration constant
f	Fitness function
g	Best swarm position
k	Iteration
N_D	Number of points in a data set
OF	Objective function
P	Pressure
R	Universal gas constant
r_1, r_2	Random numbers in PSO algorithm
s	Particle position
T	System temperature
U_{ij}	UNIQUAC parameter
v	Particle velocity
w	Inertia weight
x_i	Mole fraction in the liquid phase
y_i	Mole fraction in the vapor phase
z	Coordination number in UNIQUAC model

Abbreviations

NRTL	Non-random-two-liquid
PSO	Particle swarm algorithm
UNIQUAC	Universal-quasi-chemical
VLE	Vapor–liquid equilibrium

Greek letters

α	NRTL parameter
ε	Deviation
ϕ	Fugacity coefficient
γ	Activity coefficient

Super/subscripts

calc	Calculated
exp	Experimental
i, j	Component
L	Liquid
V	Vapor

In this work, isobaric vapor–liquid equilibrium data of binary mixtures containing alcohol + water were correlated using two activity coefficient models optimized with a PSO algorithm. Fifteen binary systems taken from the literature were selected for this study. Then, the PSO algorithm was used to determine the interaction parameters for the two models.

2. Equations of vapor–liquid equilibrium

As known, the phase equilibrium problem to be solved consists of the calculation of some variables of the set T–P–x–y (temperature, pressure, liquid-phase concentration and vapor-phase concentration, respectively), when some of them are known. For a vapor–liquid mixture in thermodynamic equilibrium, the temperature and the pressure are the same in both phases, and the remaining variables are defined by the material balance and the "*fundamental equation of phase equilibrium*" [8]. The application of this fundamental equation requires the use of thermodynamic models which normally include binary interaction parameters. These binary parameters must be determined using experimental data for binary systems. Theoretically, once these binary parameters are known one could predict the behavior of multicomponent mixtures using standard thermodynamic relations and thermodynamics models [9].

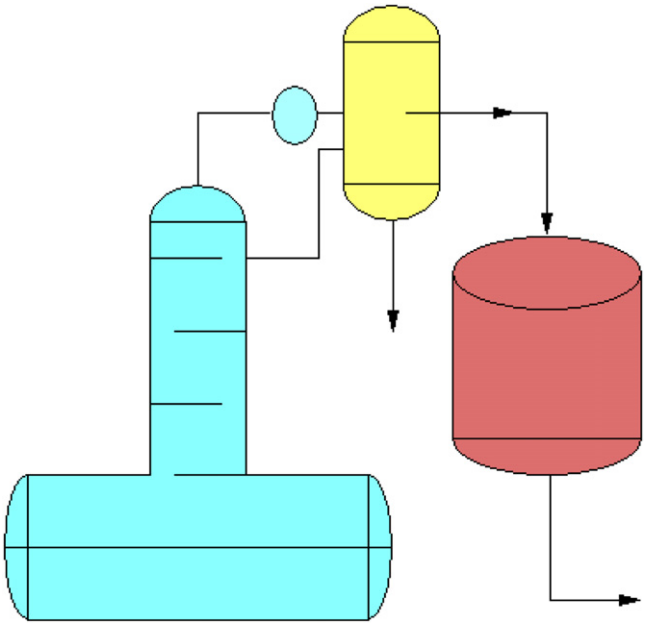

Fig. 1. Scheme of a typical batch distillation process.
Source: Taken from Ref. [6].

The fundamental equation of vapor–liquid equilibrium can be expressed as the equality of fugacities of each component in the mixture in both phases [8]:

$$\bar{F}_i^L = \bar{F}_i^V. \tag{1}$$

The fugacity of a component in the vapor phase is usually expressed through the fugacity coefficient $\bar{\phi}_i^V$:

$$\bar{F}_i^V = y_i \bar{\phi}_i^V P. \tag{2}$$

The fugacity of a component in the liquid phase is expressed through either the fugacity coefficient $\bar{\phi}_i^L$ or the activity coefficient γ_i:

$$\bar{F}_i^L = x_i \bar{\phi}_i^L P \tag{3}$$

$$\bar{F}_i^L = x_i \gamma_i f_i^0. \tag{4}$$

In these equations, y_i is the mole fraction of component in the vapor phase, x_i is the mole fraction of component in the liquid phase, and P is the pressure. The fugacity is related to the temperature, the pressure, the volume and the concentration through a standard thermodynamic relation [10].

If the fugacity coefficient is used in both phases, the method of solution of the phase equilibrium problem is known as "*the equation of state method*". Then, equation of state and a set of mixing rules are needed, to express the fugacity coefficient as a function of the temperature, the pressure and the concentration [9]. Most models available in the literature for the activity coefficient are of the correlating type (Van Laar, Margules, NRTL, UNIQUAC and Wilson) [8]. In this study, the two activity coefficient models most commonly used (UNIQUAC and NRTL), were optimized using a PSO algorithm.

The UNIQUAC model [11] is given by two contributions, a combinatorial, entropic one, related to the molecules' size and shape, and a residual, enthalpic one, related to the energy interactions. Table 1 describes the UNIQUAC activity coefficient model. In these equations, U_{ij} and U_{ji} represent the interaction energy difference between i–j and j–j pair molecules and between j–i and i–i pair molecules, respectively. R is the universal constant of gases, and T is the temperature. The required r_i and q_i parameters were calculated according to Bondi [12], and the equation below must be satisfied with the coordination number (z) equal to 10.

The NRTL model [13] for the activity coefficient is detailed in Table 2. In these equations, B_{ij} and B_{ji} represent the interaction energy difference between i–j and j–j pair molecules and between j–i and i–i pair molecules, respectively. R is the universal constant of gases, and T is the temperature.

In these models, the interaction parameters (B_{ij} and α_{ij} for NRTL, and U_{ij} for UNIQUAC) were calculated using experimental VLE data taken from the literature [10]. Fifteen isobaric VLE systems were considered. These systems are of interest for the conception of industrial operations as environmental, chemical industries, among others. The VLE data of the systems of alcohol (1) + water (2) are described in Table 3.

Table 1
Details of the UNIQUAC model.

UNIQUAC activity coefficient model
$\ln \gamma_i = \ln \gamma_i^{comb} + \ln \gamma_i^{res}$
$\ln \gamma_i^{comb} = \ln \frac{\phi_i}{x_i} + \frac{z}{2} q_i \ln \frac{\theta_i}{\phi_i} + l_i - \frac{\phi_i}{x_i} \sum_j x_j l_j$
$\ln \gamma_i^{res} = q_i \left[1 - \ln \left(\sum_j \theta_j \tau_{ji} \right) - \sum_j \frac{\theta_i \tau_{ij}}{\sum_k \theta_k \tau_{kj}} \right]$
$\phi_i = \frac{r_i x_i}{\sum_j r_j x_j} \quad \theta_i = \frac{q_i x_i}{\sum_j q_j x_j}$
$l_i = \frac{z}{2} (r_i - q_i) - (r_i - 1)$
$\tau_{ji} = \exp \left[\frac{-(U_{ij} - U_{ji})}{RT} \right]$

Table 2
Details of the NRTL model.

NRTL activity coefficient model
$\ln \gamma_i = \frac{\sum_j^N \tau_{ji} G_{ji} x_i}{\sum_k^N G_{ki} x_k} + \sum_j^N \frac{x_j G_{ij}}{\sum_k^N G_{kj} x_k} \left[\tau_{ij} - \frac{\sum_k^N \tau_{ki} G_{kj} x_k}{\sum_k^N G_{kj} x_k} \right]$
$\tau_{ij} = \frac{B_{ij} - B_{ji}}{RT} \quad G_{ij} = \exp \left(-\alpha_{ij} \tau_{ij} \right) \quad \alpha_{ij} = \alpha_{ji}$

Table 3
Details of the vapor–liquid equilibrium data considered in this study.

No.	Component (1)	T (°C)	Range x_1	Range y_1
1	Methanol	65–92	0.05–0.95	0.28–0.98
2	Ethanol	68–78	0.05–1.00	0.05–1.00
3	1-Propanol	89–95	0.05–0.95	0.35–0.87
4	2-Propanol	82–86	0.05–0.95	0.44–0.93
5	1-Butanol	94–114	0.01–0.98	0.20–0.86
6	2-Butanol	89–92	0.01–0.69	0.27–0.49
7	tert-Butanol	81–85	0.04–0.84	0.39–0.79
8	1-Pentanol	96–120	0.01–0.90	0.14–0.44
9	tert-Pentanol	87–94	0.10–0.90	0.38–0.58
10	1-Hexanol	98–134	0.60–0.95	0.08–0.42
11	2-Hexanol	97–120	0.70–0.95	0.16–0.48
12	Cyclohexanol	98–132	0.40–0.95	0.07–0.36
13	2-Methyl-1-propanol	96–107	0.01–0.98	0.15–0.93
14	Allyl alcohol	97–99	0.00–0.98	0.03–0.96
15	Benzyl alcohol	99–148	0.50–0.95	0.01–0.16

3. Optimization method and calculations

The PSO algorithm is initialized with a population of random particles and the algorithm searches for optima by updating generations [14]. In a PSO system, each particle is "flown" through the multidimensional search space, adjusting its position in search space according to its own experience and that of neighboring particles. The particle therefore makes use of the best position encountered by itself and that of its neighbors to position itself toward an optimal solution [15]. The performance of each particle is evaluated using a predefined fitness function, which encapsulates the characteristics of the optimization problem [16].

Each particle is associated with a velocity that indicates where the particle is traveling. Let k be a time instant. The new particle's position is computed by adding the velocity vector to the current position

$$s_{k+1}^i = s_k^i + v_{k+1}^i \tag{5}$$

when s and v denote a particle's position and its corresponding velocity in a search space, respectively. Being s_k^i particle i's position, $i = 1, \ldots, p$, at time instant k, the v_{k+1}^i new velocity (at time $k+1$) and p is the population size.

The velocity update equation is given by:

$$v_{k+1}^i = w_k v_k^i + c_1 r_1 \left(p_k^i - s_k^i \right) + c_2 r_2 \left(p_k^g - s_k^i \right) \tag{6}$$

where k is the current step number, w is the inertia weight, c_1 and c_2 are the acceleration constants, and r_1, r_2 are element from two random sequences in the range (0,1). The current position of the particle is determined by s_k^i; p_k^i is the best one of the solutions this particle has reached, p_k^g is the best one of the solutions all the particles have reached [17].

Step 1. Initialize algorithm. Set constants: k_{max}, v_{max}, w, c_1, c_2
Step 2. Randomly initialize the swarm positions $s_0^i \in \mathbb{R}^n$ for $i = 1, \ldots, p$.
Step 3. Randomly initialize particle velocities v_0^i for $i = 1, \ldots, p$.
Step 4. Set $k = 1$.
Step 5. Evaluate function value f_k^i using design space coordinates s_k^i:

If $f_k^i \le f_{best}^i$ then $f_{best}^i = f_k^i$, $p_k^i = s_k^i$.

If $f_k^i \le f_{best}^g$ then $f_{best}^g = f_k^i$, $p_k^g = s_k^i$.

Step 6. If stopping condition is satisfied then stop algorithm.
Step 7. Update all particle velocities v_k^i for $i = 1, \ldots, p$.
Step 8. Update all particle positions s_k^i for $i = 1, \ldots, p$.
Step 9. Otherwise set $k = k + 1$ goes to step 5.

Scheme 1. The PSO algorithm development in this study.

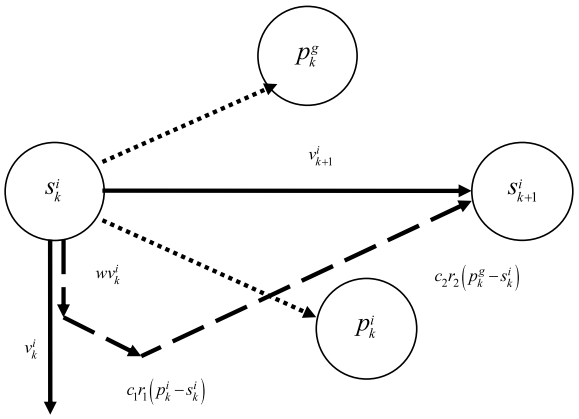

Fig. 2. PSO position and velocity update.

The variable w [18] is responsible for dynamically adjusting the velocity of the particles, so it is responsible for balancing between local and global searches, hence requiring fewer iterations for the algorithm to converge. A low value of inertia weight implies a local search, while a high value leads to a global search. Applying a large inertia weight at the start of the algorithm and making it decay to a small value through the PSO execution makes the algorithm search globally at the beginning of the search, and search locally at the end of the execution [5,17]. The following weighting function w is used in Eq. (6):

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{k_{\max}} k. \tag{7}$$

Generally, the value of each component in v can be clamped to the range $[-v_{\max}, v_{\max}]$ to control excessive roaming of particles outside the search space [5]. After calculating the velocity, the PSO algorithm performs repeated applications of the update equations above until a specified number of iterations has been exceeded, or until the velocity updates are close to zero [17]. The PSO algorithm is presented in detail in Scheme 1. Fig. 2 shows the update systems of the PSO algorithm.

The interaction parameters for each binary system: B_{ij} and α_{ij} for NRTL, and U_{ij} for UNIQUAC, were determined using a PSO algorithm programmed in MATLAB [19]. The procedure was based on the minimization of the overall objective function (OF):

$$OF = \varepsilon_y + \varepsilon_T \tag{8}$$

$$\varepsilon_y = \frac{1}{N_D} \sum_{i=1}^{N_D} \left(y_{\text{calc}} - y_{\text{exp}}\right) \tag{9}$$

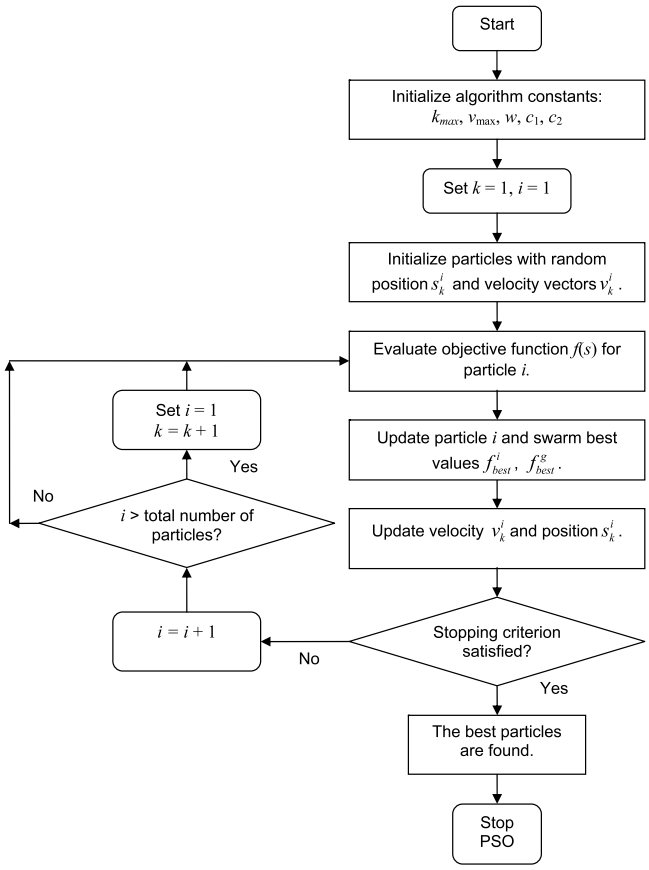

Fig. 3. Flow diagram of the PSO algorithm used in this study.

Table 4
Parameters used in the PSO algorithm.

PSO Parameter	Value
Number of particles in swarm (N_{part})	250
Number of iterations (k_{max})	1000
Cognitive component (c_1)	1.494
Social component (c_2)	1.494
Maximum velocity (v_{max})	12
Minimum inertia weight (w_{min})	0.5
Maximum inertia weight (w_{max})	0.7

$$\varepsilon_T = \frac{1}{N_D}\sum_{i=1}^{N_D}\left(T_{calc} - T_{exp}\right) \tag{10}$$

where N_D is the number of data point, T is the system temperature, y is the mol fraction in the vapor phase, and the subscripts "*exp*" and "*calc*" denote the experimental and calculated values. The summations are extended to all data points. Equilibrium temperatures and vapor phase compositions were calculated by solving the equilibrium condition given by Eq. (4). Fig. 3 shows the flow diagram of the PSO algorithm used. And Fig. 4 shows the flow diagram of the total algorithm development for the vapor–liquid equilibrium modeling.

In PSO, the inertial weight w, the constant c_1 and c_2, the number of particles N_{part} and the maximum speed of particle summary the parameters to syntonize for their application in a given problem. An exhaustive trial-and-error procedure was applied for tuning the PSO parameters. First, the effect of w was analyzed for values of 0.1 to 0.9. Fig. 5(a) shows the values of w that favored the search of the particles and accelerated the convergence. Next, the effect of N_{part} was analyzed for values of 100 to 1000 particles in the swarm. Fig. 5(b) shows that the best population to solve the problem is of 250 particles. Table 4 shows the selected parameters for the PSO algorithm.

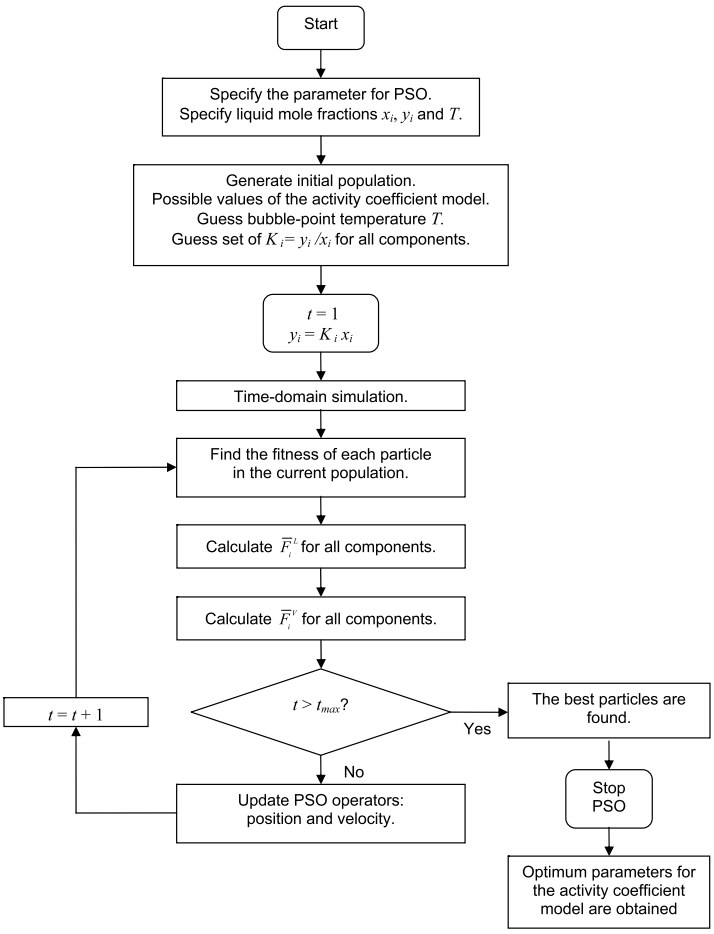

Fig. 4. Flow diagram of the total algorithm used for the vapor–liquid equilibrium modeling.

4. Results and discussion

In order to provide a substantial margin of safety, the range for the interaction parameters: B_{ij} (NRTL), and U_{ij} (UNIQUAC) was defined as $[-1000, 2500]$ cal $\cdot$ mol^{-1}. This wide range was based on physical considerations [6], and it is extremely likely that it will contain the optimal parameter values. The range for α_{ij} with theoretical bases [20] was defined as [0.2, 0.4].

Fig. 6 shows the interaction parameters: B_{ij} for NRTL and U_{ij} for UNIQUAC, determined with the PSO optimization and based on the minimization of the Eq. (8).

Table 5 shows the calculated parameter for the activity coefficient models considered (UNIQUAC and NRTL). This Table shows the deviations obtained for the Eqs. (9) and (10). The end of this Table shows the average deviation obtained by the two models.

Fig. 7 shows a comparison between the PSO algorithm development in this work and the Levenberg-Marquardt (LM) algorithm [21]. Fig. 7(a) and (b) shows the accuracy in the prediction of the vapor phase composition and the equilibrium temperatures by optimization of the UNIQUAC model. And Fig. 7(c) and (d) show the capabilities of the prediction of the vapor phase composition and the equilibrium temperatures by optimization of the NRTL model. As is observed in the figures, the best method to estimate VLE of the systems used is the PSO algorithm. This Figure ratifies the capabilities of the algorithm presented above.

5. Conclusions

In this work, two activity coefficient models (UNIQUAC and NRTL) were optimized with a PSO algorithm, and used to describe the isobaric vapor–liquid equilibrium of fifteen binary mixtures containing alcohol + water.

Based on the results and discussion presented in this study, the following main conclusions were obtained:

The PSO algorithm is appropriate to model the VLE for binary systems containing alcohol + water. The low deviations obtained with the proposed PSO algorithm indicate that can estimate the vapor phase composition and equilibrium temperatures with better accuracy than other algorithms available in the literature. The values calculated with the PSO

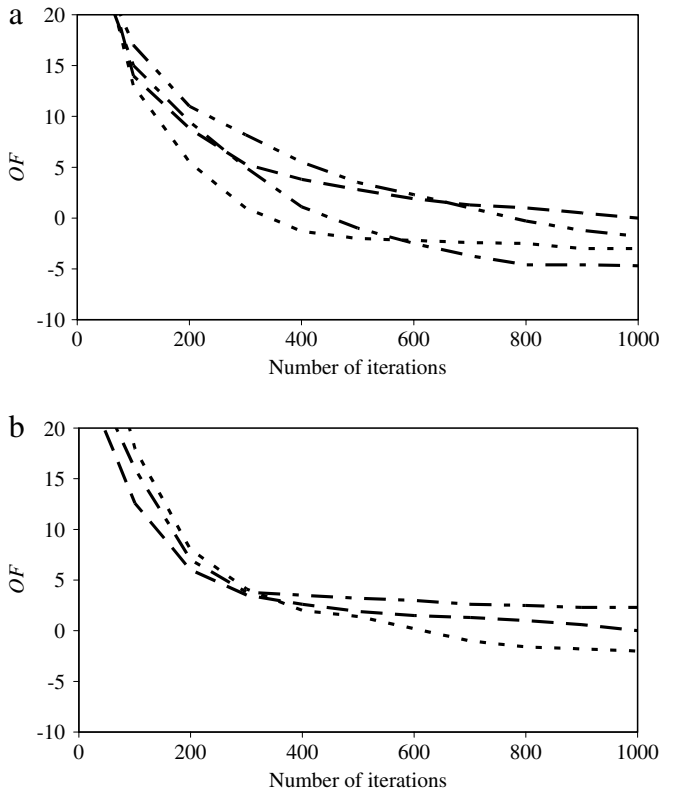

Fig. 5. Convergence graphics. (a) Determination of the best values for w as: 0.3 (— —), 0.5 (- -), 0.7 (· —), 0.9 (·· —). (b) Effect of N_{part} for: 250 (- -), 500 (— —), 1000 (· —).

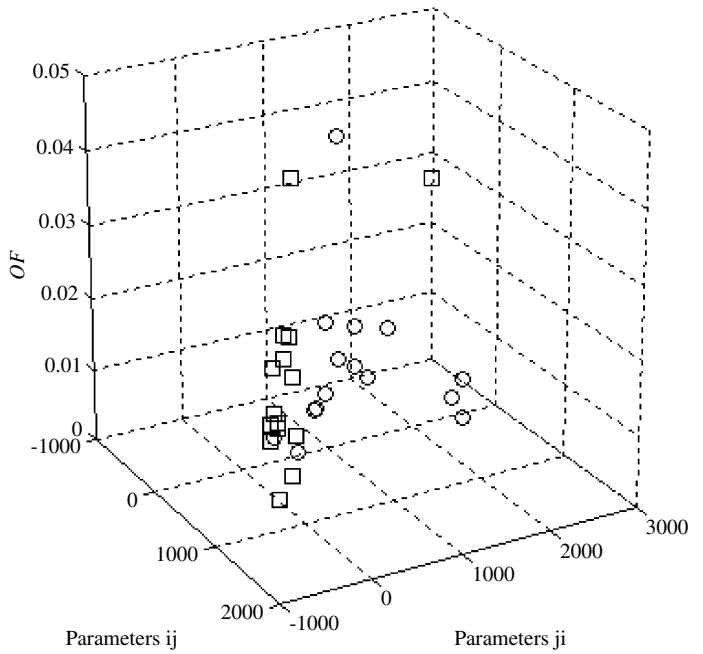

Fig. 6. Deviations of the interaction parameters estimated by minimization of the objective function. U_{i-j} for UNIQUAC (□), and B_{i-j} for NRTL (◯).

are believed to be accurate enough for engineering calculations, among other uses. Although one model (UNIQUAC and NRTL) gives better results for some particular cases, it is not possible to select one of the models as the best one for this type of optimization.

Table 5
Calculated binary interaction parameters and deviations obtained with the PSO algorithm.

System: Water (2) +	UNIQUAC model						NRTL model						
Component (1)	U_{ij}	U_{ji}	ε_T	$\varepsilon_{T(\max)}$	ε_y	$\varepsilon_{y(\max)}$	B_{ij}	B_{ji}	α_{ij}	ε_T	$\varepsilon_{T(\max)}$	ε_y	$\varepsilon_{y(\max)}$
Methanol	−392.05	652.13	0.140	0.290	0.003	0.008	−137.31	464.41	0.301	0.090	0.200	0.002	0.006
Ethanol	−196.01	769.39	0.040	0.090	0.001	0.003	−55.16	670.44	0.300	0.050	0.090	0.000	0.001
1-Propanol	183.19	273.20	0.290	1.420	0.007	0.033	17.13	947.45	0.316	0.180	1.200	0.008	0.027
2-Propanol	224.71	168.35	0.250	1.530	0.005	0.034	−3.54	859.68	0.329	0.340	1.630	0.006	0.036
1-Butanol	189.54	437.06	0.730	1.930	0.013	0.045	140.96	1349.00	0.365	0.620	2.110	0.010	0.026
2-Butanol	214.65	332.91	0.690	1.580	0.016	0.041	159.46	1179.70	0.399	0.570	1.060	0.012	0.026
tert-Butanol	437.80	54.69	0.300	1.700	0.017	0.045	28.08	950.41	0.343	0.280	1.770	0.018	0.049
1-Pentanol	237.41	369.16	0.750	1.810	0.019	0.034	18.23	1668.60	0.313	0.710	2.050	0.015	0.031
tert-Pentanol	1003.10	−128.99	0.960	2.100	0.046	0.066	328.25	846.65	0.378	0.960	2.090	0.045	0.059
1-Hexanol	810.53	−8.95	0.730	1.640	0.005	0.014	218.37	2403.80	0.296	0.914	2.140	0.007	0.030
2-Hexanol	100.62	2120.70	0.968	2.300	0.035	0.080	351.67	2174.30	0.324	0.280	0.380	0.006	0.010
Cyclohexanol	916.49	−233.22	0.620	2.050	0.013	0.030	99.55	1029.10	0.287	0.700	2.130	0.013	0.037
2-Methyl-1-propanol	313.70	249.21	0.710	2.250	0.020	0.067	187.79	1151.90	0.400	0.630	2.010	0.018	0.051
Allyl alcohol	63.53	271.21	0.180	0.330	0.006	0.014	−80.64	896.67	0.292	0.180	0.470	0.005	0.011
Benzyl alcohol	884.41	−208.10	0.901	1.840	0.003	0.008	259.15	2385.60	0.347	0.270	0.760	0.002	0.003
Average deviation			0.551		0.014					0.452		0.011	
Min. average deviation			0.040	0.090	0.000	0.000				0.050	0.090	0.00	0.000
Max. average deviation			0.970	2.300	0.050	0.080				0.960	2.140	0.04	0.060

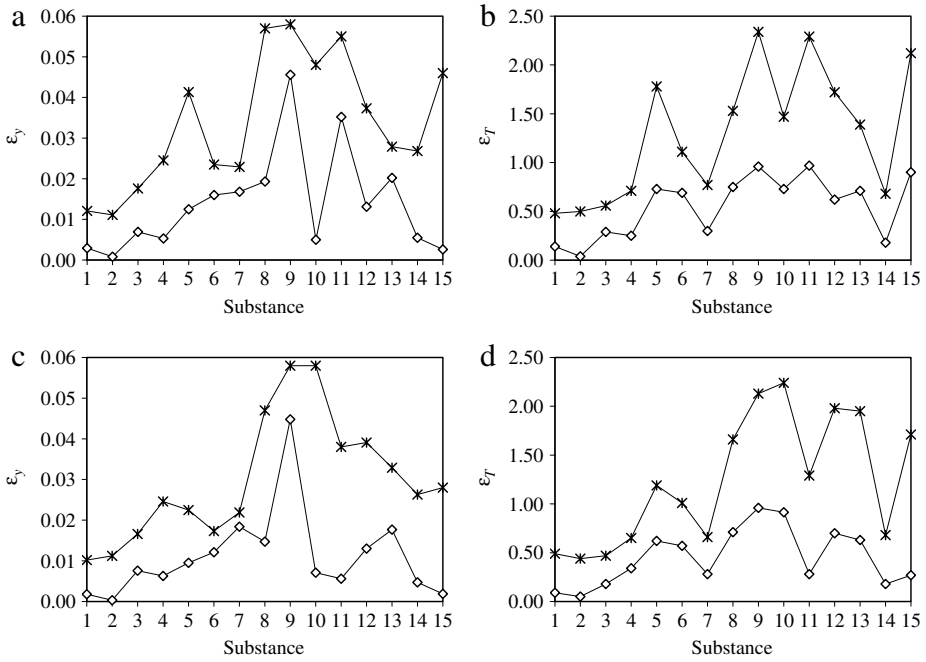

Fig. 7. Comparison between PSO (◇) and LM (Ж) optimizations used in the VLE modeling. (a) and (b) shows the deviations obtained by UNIQUAC model. (c) and (d) shows the deviation obtained by NRTL model. In this figures, the substances are listed as in Table 3.

References

[1] A. Konar, Computational Intelligence: Principles, Techniques and Applications, Springer, Berlin, Heidelberg, New York, 2005.
[2] S. Das, A. Abraham, A. Konar, Particle swarm optimization and differential evolution algorithms: technical analysis, applications and hybridization perspectives, Studies in Computational Intelligence 116 (2008) 1–38.
[3] J. Kennedy, R. Eberhart, Y. Shi, Swarm Intelligence, Morgan Kaufmann, San Francisco, 2001.
[4] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of IEEE International Conference on Neural Networks, 1995, pp. 1942–1948.
[5] J.A. Lazzús, Estimation of solid vapor pressure from CO_2+ biomolecules systems using a PSO algorithm, Journal of Supercritical Fluids 51 (2010) 312–318.
[6] Chemstations, CC-Batch in ChemCAD: Process Flowsheet Simulator, Chemstations Inc., Houston, 2001.
[7] H.Z. Kister, Distillation Design, Mc Graw Hill, New York, 1992.
[8] S.M. Walas, Phase Equilibria in Chemical Engineering, Butterworth Publishing, Storeham, 1985.
[9] H. Orbey, S.I. Sandler, Modeling Vapor–Liquid Equilibria. Cubic Equations of State and Their Mixing Rules, Cambridge University Press, USA, 1998.

[10] J. Gmehling, U. Onken, W. Arlt, Vapor–liquid equilibrium data collection. DECHEMA, Vol. 1, Part. 1, Verlag + Druckerei Friedrich Bischoff, Frankfurt, 1982.
[11] D.S. Abrams, J.M. Prausnitz, Statical thermodynamics of liquid mixtures: a new expression for the excess Gibbs energy of party or completely miscible systems, AIChE Journal 21 (1975) 116–128.
[12] A. Bondi, Physical Properties of Molecular Crystals, Liquids and Glasses, Wiley, New York, 1968.
[13] J.M. Prausnitz, R.N. Lichtenthaler, E. Gomes de Azevedo, Molecular Thermodynamics of Fluid–Phase Equilibria, Prentice Hall International Series, New Jersey, 1999.
[14] J.A. Lazzús, Estimation of density as a function of temperature and pressure for imidazolium-based ionic liquids using a multilayer net with particle swarm optimization, International Journal of Thermophysics 30 (2009) 883–909.
[15] J.A. Lazzús, Hybrid method to predict melting points of organic compounds using group contribution + neural network + particle swarm algorithm, Industrial and Engineering Chemistry Research 48 (2009) 8760–8766.
[16] J.A. Lazzús, Estimation of solid vapor pressures of pure compounds at different temperatures using a multilayer network with particle swarm algorithm, Fluid Phase Equilibria 289 (2010) 176–184.
[17] J.A. Lazzús, Prediction of sublimation pressures from SCO_2+ hydrocarbon systems using a particle swarm optimization, Journal of Engineering Thermophysics 18 (2009) 306–314.
[18] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: Proceedings of IEEE International Conference on Evolutionary Computation, 1998, pp. 69–73.
[19] MatLab, version 6.5.0, The MathWorks Inc., Natick, 2002.
[20] J. Tester, M. Modell, Thermodynamics and Its Applications, Prentice-Hall, New York, 2000.
[21] M. Reilly, Computer Programs for Chemical Engineering Education, Sterling Swift, Texas, 1972.